Maximizing the Robustness of TDMA Networks with Applications to TTP/C

Bruno Gaujal, Nicolas Navet

INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Maximizing the Robustness of TDMA Networks with Applications to TTP/C

Bruno Gaujal , Nicolas Navet

No 4614

November 7, 2002

THÈME 1

ISSN 0249-6399 ISRN INRIA/RR--4614--FR+ENG

Maximizing the Robustness of TDMA Networks with Applications to TTP/C

Bruno Gaujal * , Nicolas Navet †

Thème 1 — Réseaux et systèmes
Projet TRIO

Rapport de recherche n ° 4614 — November 7, 2002 — 31 pages

Abstract: In this study we show how one can use Fault-Tolerant Units (FTU) in an optimal way to make a TDMA network robust to bursty random perturbations. We consider two possible objectives. If one wants to minimize the probability of losing all replicas of a given message, then the optimal policy is to spread the replicas over time. This is proved using convexity properties of the loss probability. On the contrary if one wants to minimize the probability of losing at least one replica, then the optimal solution is to group all replicas together. This is proved by using majorization techniques. Finally we show how these ideas can be adapted for the TTP/C protocol.

Key-words: Real-Time Systems, Fault Tolerance, TDMA, Replica, In-Vehicle Network, TTP/C.

(Résumé : tsvp)

* ENS Lyon - LIP, 46 Allée d'Italie, 69007 Lyon, France. Email: Bruno.Gaujal@ens-lyon.fr
† LORIA, Ensem, 2 avenue de la Forêt de la Haye, 54516 Vandoeuvre, France. Email: Nicolas.Navet@loria.fr

Unité de recherche INRIA Lorraine
Technopôle de Nancy-Brabois, Campus scientifique,
615 rue de Jardin Botanique, BP 101, 54600 VILLERS L ÈS NANCY (France)
Téléphone : 03 83 59 30 30 - International : +33 3 3 83 59 30 30
Télécopie : 03 83 27 83 19 - International : +33 3 83 27 83 19
Antenne de Metz, technopôle de Metz 2000, 4 rue Marconi, 55070 METZ
Téléphone : 03 87 20 35 00 - International: +33 3 87 20 35 00
Télécopie : 03 87 76 39 77 - International : +33 3 87 76 39 77

Maximisation de la robustesse de réseaux TDMA avec application à TTP/C

Résumé : Dans cette étude, nous montrons comment utiliser de façon optimale des unités tolérantes aux fautes (Fault-Tolerant Units - FTU) pour se prémunir contre des perturbations en rafales sur un réseau TDMA. Deux objectifs sont étudiés. Si l'on veut minimiser la probabilité de perdre toutes les répliques d'un même message, la politique optimale est de répartir la transmission des répliques dans le temps. La preuve utilise des propriétés de convexité de la probabilité de perte. Au contraire, si l'on veut minimiser la probabilité de perdre une réplique ou plus, alors la solution optimale est de transmettre les répliques regroupées dans le temps. La preuve est basée sur une technique de majoration. Finalement, nous adaptons ces résultats au cas particulier du protocole TTP/C.

Mots-clé : Systèmes Temps Réel, Tolérance aux Fautes, TDMA, Station Redondante, Réseau Embarqué, TTP/C.

1 Introduction

Context of the study Multi-access protocols based on TDMA (Time Division Multiple Access) are widely used in communications systems. TDMA based protocols are particularly well suited to real-time applications since they provide deterministic access to the medium and thus bounded response times. Moreover, their regular message transmissions can be used as "heartbeats" for detecting node failures. There exists several variants of the TDMA scheme, in this paper we consider the synchronous TDMA scheme as adopted by the TTP/C protocol [20]. The stations have access to the bus in a strict deterministic sequential order, each station possesses the bus for a constant period of time called a *slot* during which it has to transmit one frame. The sequence of slots such that all stations have access once to the bus is called a *round*.

The use of TDMA based protocols is considered in high-dependability real-time applications where fault tolerance and guaranteed response times has to be provided. Examples of such applications are "brake-by-wire" and "steer-by-wire" in-vehicle applications (see [5]) or avionic applications. In such so called "X-by-wire" applications, mechanical and hydraulic components are replaced by computer control which has to be fault-tolerant. A Fault-Tolerant Unit (FTU) is a set of two or more nodes that performs the same function and thus may tolerate the failure of one or more of its constituent stations. Actually, the role of FTUs is two-fold considering the type of failure of the stations. They make the system resilient in the presence of *transmission errors* (some frames may be still be correct while others are corrupted). They also provide a way to fight against *measurement and computation errors* occurring before the transmission (some node may send the correct values while others may make errors). In the following we will see that according to which role is the most important, the optimization will lead to very different solutions.

Embedded systems may suffer from strong EMI (electro-magnetic interferences) which may represent a serious threat to the correct behavior of the system. For instance, in automotive applications, the EMI [15, 21] can either be radiated by some in-vehicle electrical devices (switches, relays..) or come from a source outside the vehicle (radio, radars, flashes of lightning..). EMI could affect the correct functioning of all the electronic devices but the transmission support is a particularly "weak link" and the use of an all-optical network, which offers very high immunity to EMI, is not generally feasible because of the low-cost requirement imposed by the industry (see [4] for more details on the electro-magnetic sensitivity of different types of transmission support). Even with a redundant transmission support, such as in TTP/C, the network is not immune to transmission errors since a perturbation is likely to affect

both channels in quite a similar manner since they are identical and very close one to each other. Unlike CAN (Controller Area Network - [9]), TDMA do not provide automatic retransmission for corrupted frames and their data are actually lost for the application.

Goal of the paper The problem we address in this study is to find the best allocation of the slot of each station in the round in such as a way as to maximize the robustness of the system against errors. We consider two distinct objectives :

1. **Objective 1 :** minimize, for each FTU, the probability that all frames of the FTU carrying the same information will be corrupted. In the rest of the paper, this probability will be termed the "loss probability" and denoted by $\mathbb{P}_{all}$.
2. **Objective 2 :** minimize, for each FTU, the probability that one (or more) frame of the FTU carrying the same information will be lost. The corresponding probability is denoted by $\mathbb{P}_{one}$.

The solution to this slot allocation problem has to take into account the fact that a data will be sent by more than one node in the same round (by all nodes of the FTU) and that it might be sent several times by a same node (in successive rounds) when the production period of the data is greater than the length of a round.

As it will be further discussed in Subsection 2.3, the two objectives correspond to well-defined situations in the field of fault-tolerance that are distinguished with regard to the concept of "fail-silence". It will also be shown that the fulfillment of these two objectives at the same time is incompatible.

Assumptions on the error model In this study, we will consider an error arrival process where "bursts" of transmission errors may occur. This is very likely in the context of in-vehicle multiplexing applications.

If successive transmission errors are not correlated (i.i.d.), it is clear that the location of each individual slot of an FTU has no influence on the loss probability since each slot has the same probability of being corrupted independently. However in practice transmission errors are highly correlated and one observes bursts or errors leading to successive transmission errors. The assumptions made for the error arrival process will thus influence the solution to the problem of locating the FTU slots. We will consider an error model that can take into account both error frequency and error gravity which generalizes a model proposed in [14]. Here are the assumptions on the perturbation errors made in the rest of the paper.

$<A_1>$ Each time an EMI occurs, it will perturb the communications on the bus during a certain duration, each bit transmitted during this perturbation has a probability π to be corrupted, independently of each other (the error model in [14] assumes $\pi = 1$).

$<A_2>$ The starting times of the EMI bursts are independent random variables, uniformly distributed over time.

$<A_3>$ The size of each EMI burst is exponentially distributed and is independent of everything else.

Without further knowledge on the considered application and its environment, assumptions $< A_1 >$ and $< A_2 >$ are rather reasonable. Assumption $<A_3>$ is more technical and will be used in the proofs of Section 3.1 (objective 1). We have some hope that the result of Section 3.1 remains valid for other distributions.

However, the results of Section 4 (objective 2) do not use assumption $<A_3>$ and are valid for all distributions of the size of the bursts (provided they remain independent of the starting point of the EMI).

Related work The Time-Triggered Architecture (TTA - see [10, 11]) has been designed for high-dependability real-time systems such as automotive applications. The TTP/C protocol [20], which is a central part of the TTA, possesses numerous features and services related to dependability such as the bus guardian [19], the group membership algorithm [17] and support for mode changes [12]. The TTA and the TTP/C protocol have been designed and extensively studied at the Vienna University of Technology. Closely related to our proposal is the work described in [7] where the reliability of the transmission on a TTP network is studied with the taking into account of transmission errors on the bus as well as failures in the TTP nodes. Under the assumption that all failures and transmission errors are statistically independent, a measure of the reliability of the transmission is given in terms of Mean Time To Failure (MTTF) where a communication failure for an FTU is defined as the loss of all messages of an FTU sent in the same round. From the MTTF of each individual FTU, a global measure of the reliability of the system is derived.

There exist two main differences with our work. One concerns the assumptions made on the perturbations and the second the data production. In [7] the errors are assumed to be independent, the location of the FTU slots has thus no influence and is not considered. Here on the contrary, we take into account the burstiness of the

perturbation process. Hence the time allocations of the FTU replicas will have a big effect on the transmission error probabilities.

As for the data production issue, in [7] failure is decided on a per round basis while in this paper this event will be assessed considering the frames sent in a production cycle of a data. Indeed, the same data might be transmitted during successive rounds and the fact that no frame of an FTU has been successfully transmitted in one round does not necessarily imply a communication failure because the same data is also sent in following rounds (see paragraph 2.2).

The second difference with [7] is that we do not merely compute the reliability of a given system but also provide a way to optimize it via time allocation of the replicas. This does not require any modification of the protocol or of the parameters of the system. Just playing with the time allocation of replicas provides a substantial gain in resilience (around 80 % in many cases) as seen in Section 3.

Finally another novelty with respect to previous work comes from the proof techniques. They are based on *multimodularity* and *Sturmian sequences* for Section 3 and on *majorization* and *Schur convexity* for Section 4. To the best of our knowledge, these notions have never been applied in this framework and they may be useful for several other related problems.

2 Framework of the study

In this section, we first describe the communication protocol, then the model of the application and the notations used. We then justify the two distinct objectives that were identified with regard to the concept of "fail-silence".

2.1 Protocol description

Throughout this paper, we will consider the synchronous TDMA protocol. The time needed to transmit one bit over the bus is taken as the time unit. In the following all time quantities are given using this time-bit as unit.

The number of *stations*, S, is static and the stations have access to the bus in a strict deterministic sequential order. Each station possesses the bus for a constant period of time called a *slot* during which it has to transmit one *frame*. The size of the slots is not necessarily identical for all stations but successive slots belonging to the same station are of the same size. The sequence of slots such that all stations have access once to the bus is called a *round*, as shown in Figure 1.

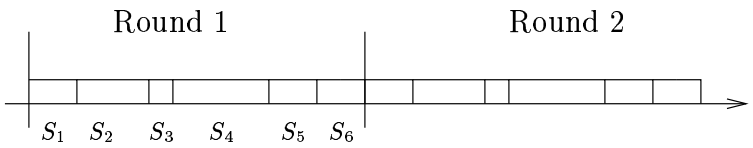

Figure 1: A round is made of S slots (here $S = 6$) , one slot per station.

2.2 Application model

To achieve fault-tolerance, that is the capacity of a system to deliver its service even in the presence of faults, some nodes are *replicated* and are clustered into *Fault-Tolerant Units* (FTUs). An FTU is a set of several stations that perform the same function and each node of an FTU possesses its own slot in the round so that the failure of one or more stations in the same FTU might be tolerated. The stations forming an FTU are called *replicas* in the following. For the sake of simplicity, a non-replicated station will also be termed an FTU (of cardinality one).

One denotes by $\mathcal{F}$ the set of FTUs : $\mathcal{F} = \{A, B, C...\}$ and C_A is the cardinality of FTU A, *i.e.* the number of stations forming FTU A. The size (in bits) of the slots of all the stations in A is the same and is denoted by h_A.

By definition, the total number of bits in a round, denoted R, is equal to:

$$R = \sum_{A \in \mathcal{F}} C_A h_A.$$

The problem consists in choosing the position of the slots of all stations forming an FTU in a round. This is done under the form of a binary vector x^A of size R (called an allocation for A) defined by

$$\forall 1 \leq i \leq R, \quad x_i^A = \begin{cases} 1 & \text{if some station in } A \text{ transmits at time-bit } i \\ 0 & \text{otherwise .} \end{cases}$$

Note that the construction of x^A must follow several constraints. First the binary vector x^A must be made of C_A "blocks" of ones, each of size h_A to correspond to an allocation of all the slots of A. Second, the allocations of all the FTUs must be *compatible*, meaning that the same bit cannot be allocated to two different FTUs. Finally all bits in a round must be allocated to some FTU. In mathematical terms these compatibility constraints can be written

$$\sum_{A \in \mathcal{F}} x^A = (1, \cdots, 1).$$

Finally, the frame sent by a node contains some data whose value is periodically updated as it is generally the case in distributed control applications. For instance, in a typical car environment, a frame sent by the engine controller may contain the RPM value plus the engine temperature and a new frame is built every 10ms (the maximum refresh rate of the two "signals" composing the frame).

Since they are replicas, all nodes of an FTU update their data with the same period denoted by T_A and called a *production cycle*. The data sent during one production cycle is also called a *message* in the following. It is also assumed that all nodes of a FTU are synchronized using the global time service requested by the communication protocol so that at each point in time each node of an FTU sends the data corresponding to the same production cycle.

The length of the TDMA round R is a function of the number of nodes, of the maximal size of the message sent in each slot, and on some characteristics of the network and of the communication controllers. The value of R is thus not generally correlated with the production period of the data [1]. If $\exists A \in \mathcal{F}$ s.t. $T_A < R$ then some data may not be transmitted which is generally unacceptable. If $\forall A \in \mathcal{F}$, $T_A > R$ then the same data is transmitted in more than one round. Also, if the beginning of the production cycle does not correspond to the beginning of a round and if FTU A has more than one replica, then data corresponding to different production cycles may be transmitted in the same round as it is the case in the first and third round of the example drawn on Figure 2.

2.3 Which objective with respect to fail-silence ?

The number of replicas per FTU which is required to tolerate k faults heavily depends on the behavior of the individual components [5]. For instance, if the failure of k nodes must be tolerated, the least necessary number of replicated nodes is $k+1$ when all nodes are *fail-silent*. A node is said fail silent if

1. a) it sends frames at the correct point of time (correctness in the time domain) and b) the correct value is transmitted (correctness in the value domain),

2. or it sends detectably incorrect frames (eg. wrong CRC) in its own slot or no frame at all.

[1] The latest version of the TTP/C specification [20] enables the designer to insert an idle time after the transmission of a frame so that the duration of a round can take an application related value. In particular it could be equal to the length of the production cycle of a data but the problem remains with data having different production cycles.

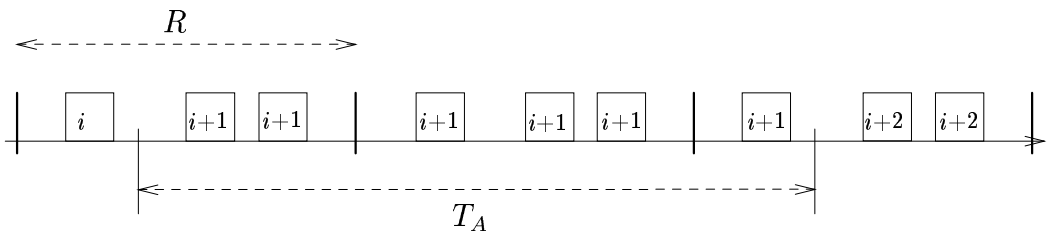

Figure 2: Three successive rounds. Only the slots allocated to the FTU A of cardinality 3 are shown. The message corresponding to the $(i+1)^{\text{th}}$ production cycle is sent over 3 rounds.

TTP/C provides very good support for the requirements 1.a) and 2) (whose fulfillment provide the so-called "fail-silence in the temporal domain") especially through the bus guardian concept while the value domain is mainly the responsibility of the application. The reader is referred to [5, 19, 18] for good starting points on the problem of ensuring fail-silence. For FTUs composed of a set of fail-silent nodes, the successful transmission of one single frame for the whole set of replicas is sufficient since the value carried by the frame is necessarily correct. In this case, the main objective to achieve with regard to the robustness against transmission errors is the minimizing of $\mathbb{P}_{all}$, that is the probability that all frames of the FTU (carrying data corresponding to the same production cycle) will be corrupted.

In practice replicated sensors may return slightly different observations and, without extra communication for an agreement, replicated nodes of a same FTU may transmit different data. If a decision, such as a majority vote, is taken by a consumer node with regard to the value of the transmitted data, the objective is to minimize $\mathbb{P}_{one}$ the probability that one (or more) replica of a message will be lost.

From an implementation point of view, it is generally preferable to present only one copy of a data to the application in order to simplify the application code and to keep it independent of the level of redundancy (i.e. number of nodes composing the FTU). In OSEK/VDX[2] terminology, the algorithm responsible for the choice of the value that will be transmitted to the application is termed "the agreement algorithm". It is usually necessary that the agreement algorithm works with consistent data that is data corresponding to the same cycle of production or sampled at the

[2]OSEK/VDX is a project of the automotive industry aiming at building standard architecture for in-vehicle control units. Detailed information can be obtained at http://www.osek-vdx.org .

same point in time. Many agreement strategies are possible : pick-any (fail-silent node), average-value, pick-a-particular-one (the selected value has been produced by the best sensor), majority vote etc ... A proposal of a software layer responsible for implementing the agreement strategy has been made by the OSEK/VDX consortium [16].

3 Minimizing the loss probability

If one wants to build the system in such a way as to minimize the loss probability, one must not tackle the problem on a per round basis but indeed consider the probability that all frames from the same FTU in one production cycle are corrupted, because this means that one data is completely lost.

In this section, we investigate the problem of minimizing the loss probability $\mathbb{P}_{all}$. In Subsection 3.1, we focus on the optimal policy for one FTU. In Subsection 3.2 we consider all FTU combined and an heuristic is proposed for the case where more than two different cardinalities coexist. Its performances are assessed by simulation.

3.1 Optimal allocation of a single FTU

Here we focus on a given FTU, say A made of $K := C_A$ replicas per round, all of size $h := h_A$. The problem here is to find an allocation x of the K replicas over one round that minimizes the probability $\mathbb{P}_{all}$ that all replicas carrying the same message are lost, regardless of the other FTUs.

The proof technique uses two notions introduced in [2], multimodularity and Sturmian sequences.

3.1.1 Optimization using multimodularity and Sturmian sequences

Let x be a binary vector of size R. Its *density* is $(1/R)\sum_{i=1}^{R} x_i$. A binary vector is a *block-vector* with blocks of size h if $x_i = 1$ only in intervals of h consecutive values. A *block shift* is a vector δ_i such that $\delta_i(n) = 0$ for all n except $\delta_i(i) = +1$ and $\delta_i(i+h) = -1$. Basically if x is a block-vector, then $x + \delta_i$ is also a block-vector similar to x with one of its blocks shifted to the left by one unit as in the following example with blocks of size 3.

$$\begin{aligned} x &= (0,0,0,0,1,1,1,0,1,1,1,0,0,1,1,1,0) \\ x+\delta_4 &= (0,0,0,1,1,1,0,0,1,1,1,0,0,1,1,1,0). \end{aligned}$$

A *global shift of size* j , s_j is an operation on vectors that shifts all values to the left by j (modulo the size of the vector) as in the following example.

$$\begin{aligned} x &= (0,0,0,0,1,1,1,0,1,1,1,0,0,1,1,1,0) \\ s_2(x) &= (0,0,1,1,1,0,1,1,1,0,0,1,1,1,0,0,0). \end{aligned}$$

A real function $F(x)$ is *block-multimodular* with blocks of size h if the following inequality holds for all block-vectors x.

$$\forall i \neq j \quad F(x+\delta_i) + F(x+\delta_j) \geq F(x) + F(x+\delta_i+\delta_j),$$

as soon as $x+\delta_i, x+\delta_j, x+\delta_i+\delta_j$ are all block vectors.

A *Sturmian sequence* v with density a/b is a binary vector of size b such that

$$v_n = \lfloor na/b \rfloor - \lfloor (n-1)a/b \rfloor. \tag{1}$$

For example, the Sturmian sequence with density $3/8$ is $v_{3/8} = (0,0,1,0,0,1,0,1)$.

A *block Sturmian vector* x with density $ha/(b+(h-1)a)$ with blocks of size h is constructed from v in the following way.

- Start with x empty.
- If $v_i = 1$, then $x := x.1\cdots 1$, (with h ones concatenated at the end of x).
- If $v_i = 0$, then $x := x.0$.

Continuing the example, the block Sturmian vector with blocks of size 3 with density $9/14$ is derived from $v_{3/8}$ using the procedure above:

$$x = (0,0,1,1,1,0,0,1,1,1,0,1,1,1).$$

Note that x is not equal to $v_{9/14}$, the Sturmian sequence with density $9/14$, since $v_{9/14}$ does not contain blocks of size 3: $v_{9/14} = (0,1,0,1,1,0,1,1,0,1,1,0,1,1)$. For more details on multimodularity and Sturmian sequence, the reader might refer to [8, 2].

One can apply a general optimization Theorem given in [2] to the block case. This theorem relates the minimizing of multimodular functions with Sturmian vectors.

Theorem 1. *[2] If F is a block-multimodular function, then consider the shift invariant version of F, namely the function $G(x) := 1/R \sum_{i=0}^{R-1} F(s_i(x))$. Then G is minimized over all block vectors with density d by the block Sturmian vector of density d.*

3.1.2 Multimodularity of the loss probability

Here, we prove that the loss probability $\mathbb{P}_{all}$ is block-multimodular. Let us first make the following assumption :

$<A_4>$ In the initialization phase, we assume that the start of the first production cycle is uniformly distributed over the first round.

ii We consider a single error burst that begins at a time uniformly distributed over the past and whose size is exponentially distributed with parameter λ.

Lemma 1. *Consider a single error burst that begins at a time uniformly distributed over the past and whose size is exponentially distributed with parameter λ. Under and assumptions $<A_1>$, $<A_3>$ and $<A_4>$ above, the probability $\mathbb{P}_{all}$ of losing all replicas of FTU A is block multimodular, with blocks of size h_A.*

Proof. We consider x an arbitrary allocation for A. We look at the probability that an error corrupts all replicas carrying a given message m for allocation x. The same message (m) is emitted by a number of replicas which can be written as NK where N is an integer, and $K := C_A$ is the number of replicas per round. For notation simplicity, we also set $h := h_A$.

In the following we also denote by C the round where message m begins. We denote by P_k the position of the last bit of the k-th replica for the FTU A in x and by d_k^i the "distance" between replica k and replica $k+i$: $d_k^i = P_{k+i} - P_k$. We denote by $\mathbb{P}_{all}(x)$ the loss probability of m under allocation x; by $\mathbb{P}_0(x)$ the loss probability under allocation x given that the perturbation starts in a round preceding round C and by $\mathbb{P}_1(x)$ the loss probability under allocation x given that the perturbation starts in the same round (C). When the EMI burst covers the whole message, there is a relation between the random variables corresponding to the beginning of the message m (called B) and the beginning of the error burst (called S) respectively. Basically, the error must start before the end of the first replica carrying message m.

See Figures 3 and 4 for an illustration of cases $\mathbb{P}_0$ and $\mathbb{P}_1$ respectively.

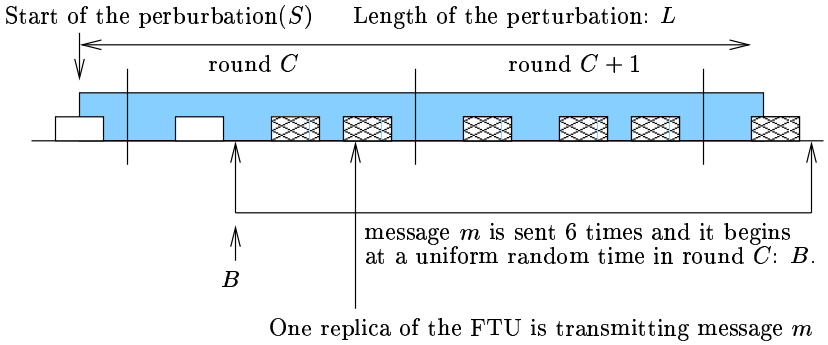

Figure 3: A perturbation burst which begins in a round preceding the start of a message covers the whole message (case $\mathbb{P}_0$).

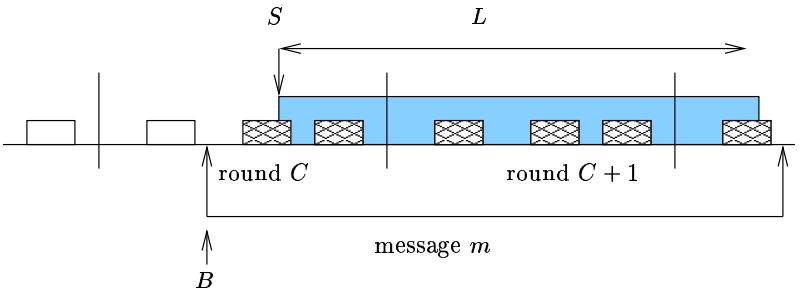

Figure 4: A perturbation burst, beginning in the same round as the message, covers the whole message (case $\mathbb{P}_1$).

By conditioning over the values of B and L (which are independent variables), we obtain :

$$\mathbb{P}_1(x) = \sum_{k=CK+1}^{CK+K} \Big[(1-(1-\pi)^h)^{NK} Pr(P_{k-1} - h < B < P_k - h) Pr(CR \leq S \leq P_k)$$
$$Pr(L + S \geq RN - d^1_{k-1} - h + P_k) \Big]$$
$$= \frac{(1-(1-\pi)^h)^{NK}}{R^2} \sum_{k=CK+1}^{CK+K} d^1_{k-1} \int_{CR}^{P_k} \exp(\lambda(-RN + d^1_{k-1} + h - P_k + S)) dS$$
$$= \frac{(1-(1-\pi)^h)^{NK}}{\lambda R^2} \sum_{k=1}^{K} d^1_{k-1} \exp(\lambda(-RN + d^1_{k-1} + h))(1 - \exp(-\lambda P_k)),$$

and

$$\mathbb{P}_0(x) = \sum_{k=CK+1}^{CK+K} (1-(1-\pi)^h)^{NK} Pr(P_{k-1} - h < B < P_k - h) Pr(L + S \geq RN + P_k - d^1_{k-1} - h)$$
$$= \frac{(1-(1-\pi)^h)^{NK}}{R} \sum_{k=CK+1}^{CK+K} d^1_{k-1} Pr(L + S \geq RN + P_k - d^1_{k-1})$$
$$= \frac{(1-(1-\pi)^h)^{NK}}{R} \sum_{k=CK+1}^{CK+K} d^1_{k-1} \int_0^{CR} \exp(\lambda(-RN - P_k + S + d^1_{k-1})) dS/CR$$
$$= \frac{\pi^{NK}(1-(1-\pi)^h)^{NK}}{\lambda CR^2} \sum_{k=1}^{K} d^1_{k-1} \exp(-\lambda RN - \lambda P_k + \lambda d^1_{k-1})(1 - \exp(-\lambda CR)).$$

Finally,

$$\begin{aligned}
\mathbb{P}_{all}(x) &= 1/(C+1)\mathbb{P}_1(x) + C/(C+1)\mathbb{P}_0(x) \\
&= M \sum_{k=1}^{K} d^1_{k-1}((1 - \exp(-\lambda P_k)) + \exp(-\lambda P_k)(1 - \exp(-\lambda CR))) \exp(\lambda d^1_{k-1}) \\
&= M \sum_{k=1}^{K} d^1_{k-1}(1 - \exp(-\lambda P_k - \lambda CR)) \exp(\lambda d^1_{k-1}),
\end{aligned}$$

where

$$M = \frac{\exp(-\lambda RN + \lambda h)(1-(1-\pi)^h)^{NK}}{(C+1)\lambda R^2}.$$

We consider shifts to the left of the normalization $P(x) := \mathbb{P}_{all}(x)/M$.

$$\begin{aligned} P(x+\delta_i) &= \sum_{k=1,k\notin\{a,a+1\}}^{K} d^1_{k-1}(1-\exp(-\lambda P_k - \lambda CR))\exp(\lambda d^1_{k-1}) \\ &+ (d^1_{a-1}-1)(1-\exp(-\lambda P_a + \lambda - \lambda CR))\exp(\lambda d^1_{a-1} - \lambda) \\ &+ (d^1_a+1)(1-\exp(-\lambda P_{a+1} - \lambda CR))\exp(\lambda d^1_a + \lambda). \end{aligned}$$

We need to distinguish the cases where $|b-a| = 1$. We focus on the case where $b = a+1$ (the case $a = b+1$ is symmetrical by exchanging the roles of a and b).

If $b > a+1$,

$$\begin{aligned} P(x+\delta_i+\delta_j) &= \sum_{k=1,k\notin\{a,a+1,b,b+1\}}^{K} d^1_{k-1}(1-\exp(-\lambda P_k - \lambda CR))\exp(\lambda d^1_{k-1}) \\ &+ (d^1_{a-1}-1)(1-\exp(-\lambda P_a + \lambda - \lambda CR))\exp(\lambda d^1_{a-1} - \lambda) \\ &+ (d^1_a+1)(1-\exp(-\lambda P_{a+1} - \lambda CR))\exp(\lambda d^1_a + \lambda) \\ &+ (d^1_{b-1}-1)(1-\exp(-\lambda P_b + \lambda - \lambda CR))\exp(\lambda d^1_{b-1} - \lambda) \\ &+ (d^1_b+1)(1-\exp(-\lambda P_{b+1} - \lambda CR))\exp(\lambda d^1_b + \lambda). \end{aligned}$$

If $b = a+1$, on the other hand, we get

$$\begin{aligned} P(x+\delta_i+\delta_j) &= \sum_{k=1,k\notin\{a,a+1,a+2\}}^{K} d^1_{k-1}(1-\exp(-\lambda P_k - \lambda CR)))\exp(\lambda d^1_{k-1}) \\ &+ (d^1_{a-1}-1)(1-\exp(-\lambda P_a + \lambda - \lambda CR))\exp(\lambda d^1_{a-1} - \lambda) \\ &+ (d^1_a)(1-\exp(-\lambda P_{a+1} + \lambda - \lambda CR))\exp(\lambda d^1_a) \\ &+ (d^1_{a+1}+1)(1-\exp(-\lambda P_{a+2} - \lambda CR))\exp(\lambda d^1_{a-1} + \lambda). \end{aligned}$$

If we compute $Q := P(x+\delta_i) + P(x+\delta_j) - P(x+\delta_i+\delta_j) - P(x)$, we get 0 when $b > a+1$ and when $b = a+1$, we get

$$\begin{aligned} Q &= (d_a^1+1)(1-\exp(-\lambda P_{a+1} - \lambda CR))\exp(\lambda d_a^1 + \lambda) \\ &- (d_a^1)(1-\exp(-\lambda P_{a+1} + \lambda - \lambda CR))\exp(\lambda d_a^1) \\ &+ (d_a^1-1)(1-\exp(-\lambda P_{a+1} + \lambda - \lambda CR))\exp(\lambda d_a^1 - \lambda) \\ &- d_a^1(1-\exp(-\lambda P_{a+1} - \lambda CR))\exp(\lambda d_a^1). \end{aligned}$$

After some simplifications, we get

$$\begin{aligned} Q &= \exp(\lambda(d_a^1+1))d_a^1 + d_a^1\exp(\lambda(d_a^1-1)) - 2d_a^1\exp(\lambda d_a^1) \\ &+ \exp(\lambda(d_a^1+1)) - \exp(\lambda(d_a^1-1)) \\ &+ \exp(\lambda(d_a^1 - P_{a+1} - RC)) - \exp(\lambda(d_a^1 - P_{a+1} - RC + 1)) \\ &\geq 0. \end{aligned}$$

The first line is non-negative by convexity of the function $z \mapsto \exp(\lambda z)$. The sum of the second and third lines is also non-negative by convexity of the function $z \mapsto \exp(\lambda z)$. □

The proof calls for several comments.

- We assumed that the same message is sent during an integer number of rounds. In fact, the proof can be adapted for the case where a message only covers a fractional number of rounds (*i.e.* of the form $NK + t$, with t smaller than K).
- We believe that the multimodularity holds for more general distributions of the error size. However we checked that it does not hold for Pareto distributions. The exponential distribution assumption is also crucial in the proof the next theorem.

Theorem 2. *Under the assumptions $<A_1>$,$<A_2>$,$<A_3>$ and $<A_4>$, the probability $\mathbb{P}_{all}$ of losing all replicas of FTU A forming the same message is minimized if the replicas are allocated over each round according to a block Sturmian sequence.*

Proof. By considering only bursts between time 0 and round C, we can assume using $< A_2 >$ that all the bursts start at independent random times, uniformly distributed. The fact that each individual burst is of exponential size ($< A_3 >$), makes is possible to discard overlaps.

The second step of the proof consists in noticing that $\mathbb{P}_{all}$ does not depend on shifts of the allocation sequence x. This means it is equal to its shift invariant version.

Finally, Theorem 1 together with Lemma 1 (which is now true for each burst independently), show that the function $\mathbb{P}_{all}$ is block-multimodular and is minimized if the allocation of the replicas form a block Sturmian sequence. □

In order to illustrate this result, we shall develop a small example. Let us consider a case with an FTU A made of 4 replicas of size 2 bits over a round with a total of 15 bits. then one optimal allocation of A is to allocate the replicas using a block Sturmian sequence with blocks of size 2, in positions $P_1 = 1, P_2 = 5, P_3 = 9, P_4 = 13$, as shown in Figure 5.

Figure 5: Block Sturmian allocation of size 2 of all replicas of A.

This allocation of the replicas is the best to fight against EMI satisfying assumptions $<A_1>$,$<A_2>$ and $<A_3>$.

3.2 Slot allocation of all FTUs

In this section, we consider all FTUs together and we try to find an allocation of all of them simultaneously. An optimal allocation for each FTU constructed using theorem 1 is not always feasible since the allocations may be conflicting with each other (if two allocations have at least one bit in common).

In the following we distinguish the case where it is possible to allocate all FTUs optimally and the case where this is not possible and where compromises have to be found.

3.2.1 Some optimal cases

Assume that the replicas of all FTUs have the same size (h). When the number of replicas per FTU takes up to two different values, it is possible to allocate the slots for all FTUs according to a block Sturmian sequence for each of them with no conflicts

Here is the way to build an allocation which is a block Sturmian vector[3] for all FTUs.

[3]or a shift of a block Sturmian vector which performs the same.

Let n be the total number of FTUs ($\#\mathcal{F} = n$). Assume that the cardinality of all FTUs is either c or d. The FTUs with cardinality c (resp. d) are denoted $A_1, \cdots, A_m$ (resp. $B_1, \cdots, B_\ell$), with $m + \ell = n$.

1. Build a Sturmian sequence with density $cm/(cm + d\ell)$.
2. Replace all the "1"'s (resp. "0"'s) by $A_1, \cdots A_m$ (resp. $B_1, \cdots, B_\ell$) in a round-robin fashion.

What is finally obtained is a sequence over all FTUs $A_1, \cdots A_m, B_1, \cdots, B_\ell$ with block Sturmian allocation for all of them.

To illustrate this algorithm, let us construct an example. We consider a round made of 5 FTUs: A, B, C, D, E, with respective cardinalities 5,2,5,2,2. We will build an allocation which is Sturmian for each one of them and has no conflict. First we construct a Sturmian sequence with density 10/16, this is 0101101101011011. Now we replace each "1" by A or C (in a round-robin manner) and each "0" by D or E or B (again in a round-robin manner). We get the following optimal allocation:

D	A	E	C	A	B	C	A	D	C	E	A	C	B	A	C

In addition, there exist some cases where the cardinalities take more than two values and a Sturmian allocation is still possible for all FTUs. If the cardinalities are of the form $1, 2, 4, 8, 16, \cdots, 2^k$ (all powers of 2) then, it is possible to find Sturmian allocations for all FTUs. The Fraenkel conjecture (see [1]) says that these are essentially the only cases where the superposition of several Sturmian allocations is possible without conflicts.

At this point, we should point out that the case with up to two different cardinalities should fulfill most of the needs. In a system where only a subset of nodes are critical from a the point of view of the dependability, FTUs will generally be of cardinality one (non-critical nodes) and two (critical nodes). In the context of X-by-Wire applications where dependability constraints are stringent, two different cardinalities should also generally be sufficient. For instance the prototype designed in the Brite Euram III project "Safety related Fault Tolerant Systems In Vehicle" (see [5]) is composed of nodes of cardinalities two (the steering wheel actuator and the steering control unit) and three (steering actuators).

3.2.2 General case

As mentioned before, the case with two different cardinalities is rather common in practice. Nevertheless, it could happen that a more difficult configuration arises. In

general, it is not possible to allocate the slots of all FTUs according to Sturmian sequences without getting conflicts. Two possible strategies can be considered:

1. One can deliberately favor a subset S of particularly critical FTUs having all the same cardinality K and the same size h. In this case, the slots of those FTUs are allocated optimally (regarding the loss probability) while the slots of the others FTUs are fit in the remaining free places. The allocation is given by any block Sturmian sequence (see equation 1) of density $\alpha = \#(S)Kh/R$ as done in the previous section.

2. No FTUs are of special importance and a solution minimizing the loss probability for the set of all messages of the system has to be found.

3.2.3 Heuristic solution to the allocation problem

In this paragraph, we propose a low complexity heuristic solution for the combined allocation problem and assess its performance against random allocations under various perturbation conditions. The performance metric is the loss probability $\mathbb{P}_{all}$. As for a Sturmian allocation, the idea of this heuristic is to spread the replicas of a same FTU as evenly as possible over time.

For each FTU A with cardinality C_A and frames of size h_A, we define the density of frames per bit: $u^A := C_A h_A/R$. Intuitively, u^A is the number of frames belonging to FTU A that should be transmitted per bit. The sum of the densities up to bit k for FTU A is $U_k^A := ku^A$. We denote by n_i^A the number of bits FTU A has already been allocated up to step i (including step i). At each step, an FTU will be allocated the number of bits necessary to send its frame. In the following, $s(i)$ indicates the FTU chosen at step i while $b(i)$ is the total number of bits already allocated at step i.

1. Initialization step : $n_0^A := 0$, $b(0) := 0$ and $i := 1$.

2. At step i the FTU for which the difference between the number of "due" bits and the previous allocation is maximum is selected:

$$s(i) := \text{argmax}_{A \in F}(U^A_{b(i-1)+1} - n^A_{i-1}).$$

3. The next $h_{s(i)}$ bits are allocated for FTU $s(i)$.

4. Perform the updates $b(i) := b(i-1) + h_{s(i)}$, $n_i^{s(i)} := n_{i-1}^{s(i)} + h_{s(i)}$ and $n_i^A := n_{i-1}^A$ if $A \neq s(i)$.

5. if $b(i) = R$ stop else i := i+1, go to item 2.

The algorithmic complexity of the heuristic allocation is linear in the number of bits of a round. Note that a similar construction based on density has been successfully used for defining a policy that shapes real-time traffic in [6].

3.2.4 Performance comparison with random allocations

To assess the robustness of the allocations given by the heuristic, simulations were performed against random allocations. A configuration is defined by a number of FTU and the cardinality of each FTU. We distinguish two classes of problem according to the number of FTUs on the network: for a "medium size problem" there are at least 3 FTUs and at most 6 FTUs while in a "large size problem" there are up to 12 FTUs. Two hundreds configurations were randomly generated with FTUs having a cardinality between 2 and 4. For each configuration, we randomly pick up 100 hundred slots (in the 1000 first rounds) where a data is transmitted for the first time. The duration of the production cycle of the data is equal to 3 rounds and is denoted by T. Then for each selected start of transmission, 500 bursts of errors are generated with $\pi = 1$ and a size exponentially distributed of mean $c \cdot T$ with $c \in \{0.5, 1, 1.5, 2\}$. If the burst of errors starts before the end of transmission of the first replica and finishes after the start of transmission of the last replica, the data is lost. The results of these experiments are shown on Figure 6.

The use of the proposed heuristics greatly diminishes the total number of lost data (up to 79%) knowing that there are cases where the size of the burst is such that the data cannot be transmitted whatever the allocation. This fact explains why the efficiency of the heuristic tends to be lower when the size of the burst is becoming larger. It is also noteworthy that the heuristic is robust to an increase of the size of the burst since the performance on the "large size problem" is close to the one obtained on the "medium size problem".

3.2.5 Performance compared with optimal allocations

We now evaluate the behavior of the heuristic with regard to the optimal Sturmian allocation. We consider a case with only two replica cardinalities. Using the previous section, we know that we can construct on optimal allocation. The heuristic allocation will not necessarily find this optimal allocation and we want to measure how well it performs compared to the optimal.

We consider 200 random configurations of the medium size problem for which the optimal allocation is known (ie. number of FTUs cardinalities is less than 3).

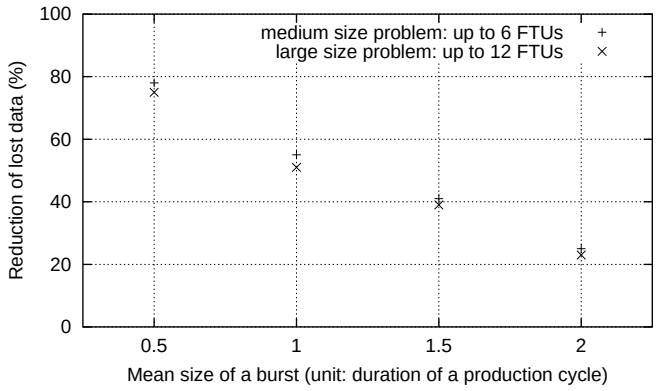

Figure 6: Reduction of the number of lost data when the heuristic is used instead of a random allocation. The mean burst size ranges from 0.5 to 2 times the length of a production cycle which is chosen equal to 3 TDMA rounds.

The conditions of the experiment are the same as in paragraph 3.2.4 except that the number of first transmission slots that are selected is equal to 1000 (in the first 2000 rounds) and that 5000 bursts of errors are randomly generated. The loss of performance against the optimal solution is shown on Figure 7.

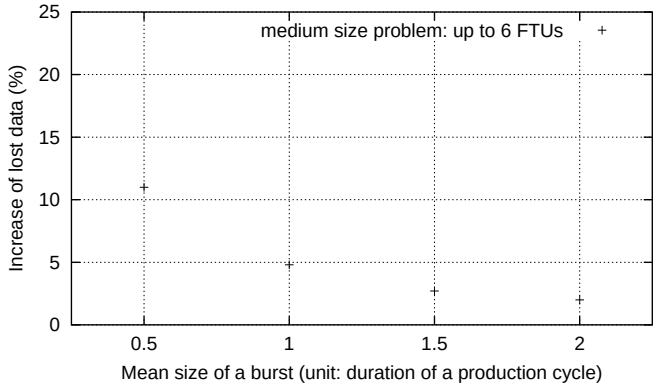

Figure 7: Increase of the lost data when the heuristic is used instead of the optimal allocation.

The loss of performance with regard to the optimal solution is small (less than 11% on this experiment) and it logically decreases when the size of the bursts becomes larger. The good behavior of the heuristic on configurations with less than 3 different cardinalities is a positive element with regard to its performance on arbitrary configurations.

4 Minimizing the probability that at least one replica is corrupted

Unlike the previous case, the technique used to find the optimal allocation of the replicas of one FTU is based on majorization and Schur convexity.

4.1 Schur convexity and majorization

Let $u = (u_1, \cdots, u_n)$ and $v = (v_1, \cdots, v_n)$ be two real vectors of size n. We denote by $(u_{[1]} \cdots, u_{[n]})$ and $(v_{[1]}, \cdots, v_{[n]})$ the permutations of u and v such that $u_{[1]} \leq \cdots \leq u_{[n]}$ and $v_{[1]} \leq \cdots \leq v_{[n]}$. The vector u *majorizes* v ($u \succ v$) if the following conditions hold:

$$\sum_{i=1}^{n} u_i = \sum_{i=1}^{n} v_i, \tag{2}$$

$$\sum_{i=1}^{k} u_{[i]} \leq \sum_{i=1}^{k} v_{[i]}. \tag{3}$$

A function f from $\mathbb{R}^n$ to $\mathbb{R}$ is *Schur convex* (resp. *Schur concave*) if $u \succ v$ implies $f(u) \geq f(v)$ (resp. $f(u) \leq f(v)$). For more details on these notions, the reader can refer to [13].

4.2 Schur concavity of $\mathbb{P}_{one}$

In this section we will show that the probability that the an error burst corrupts at least one replica within a production cycle ($\mathbb{P}_{one}$) is a Schur concave function with respect to the allocation of the replicas. Using the definition of Schur concavity, this will provide directly the best allocation minimizing $\mathbb{P}_{one}$.

We denote by $t = NK$ the number of frames (of size h) composing a message for FTU A. Let x be an allocation of the K replicas forming FTU A. The quantity $I_i(x)$ denotes the interval between the end of replica r_{i-1} and the beginning of replica r_i.

We denote by $I(x)$ the sequence of intervals $(I_1, \cdots I_t)$ and by $|I(x)|$ the vector of the length of the intervals, $|I(x)| = (|I_1|, \cdots, |I_t|)$. Note that $|I_1(x)| + \cdots + |I_t(x)| = N(R - Kh)$ does not depend on the allocation x.

Lemma 2. *Let us consider a single error burst starting at a random time uniformly distributed over one round and assume $\pi = 1$. Let x and x' be two allocations of A. If $|I(x)| \prec |I(x')|$ then the probabilities of losing at least one frame satisfy $\mathbb{P}_{one}(x) \geq \mathbb{P}_{one}(x')$.*

Proof. First note that bursts starting during the transmission of one replica of A corrupts this replica with probability 1, because $\pi = 1$. This means that the difference between $\mathbb{P}_{one}(x)$ and $\mathbb{P}_{one}(x')$ can only come from errors starting in between the replicas.

Also note that if $t = 1$ then $|I(x)| = |I_1(x)| = N(R - Kh) = |I_1(x')| = |I(x')|$ and all allocations are equivalent since the error model is time homogeneous.

If $t \geq 2$, we renumber the intervals of x and x' such that $|I_{[1]}| \leq \cdots \leq |I_{[t]}|$ and $|I'_{[1]}| \leq \cdots \leq |I'_{[t]}|$. Using the majorization condition, one gets for all j, $\sum_{i=1}^{j} |I_{[i]}| \geq \sum_{i=1}^{j} |I'_{[i]}|$.

We now prove by induction that for all $1 \leq j \leq t$ one can construct a coupling between $I_{[1]}, \cdots, I_{[j]}$ and $I'_{[1]}, \cdots, I'_{[j]}$ such that the probability $\mathbb{P}'_j$ that an error starting in $I'_{[1]}, \cdots, I'_{[j]}$ and corrupting at least one replica is smaller than the corresponding probability $\mathbb{P}_j$ in $I_{[1]}, \cdots, I_{[j]}$. For $j = 1$, the coupling is done according to Figure 8.

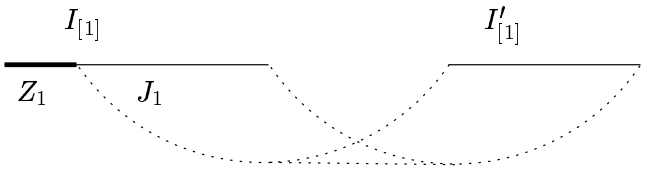

Figure 8: Coupling for the smallest interval.

After the coupling, the interval $I_{[1]}$ is split into two intervals, Z_1 and J_1 such that $I_{[1]} = Z_1 \cup J_1$ and $|I'_{[1]}| = |J_1|$. A burst starting in J_1 has the same probability of corruption that a burst starting in $I'_{[1]}$, because it is enough that the burst overlaps the interval to corrupt the replica that follows. The remaining zone (Z_1) is such that an error starting in Z_1 corrupts one replica with a non-negative probability. Therefore, $\mathbb{P}_1 \geq \mathbb{P}'_1$.

The proof continues by induction on j. The induction property is that for a given j one can construct a splitting of $I_{[1]}, \cdots, I_{[j]}$ into $(J_1, Z_1), \cdots, (J_j, Z_j)$ such

that the probability that a burst starting in $J_1 \cup \cdots \cup J_j$ is larger or equal than in $I'_{[1]} \cup \cdots \cup I'_{[j]}$ and the zone $Z_1 \cup \cdots \cup Z_j$, has a non-negative total probability of corrupting a replica.

We now add $I_{[j+1]}$ and $I'_{[j+1]}$. Two cases can occur.

1) If $I_{[j+1]} \geq I'_{[j+1]}$ then one splits $I_{[j+1]}$ as it has been done for $I_{[1]}$ and $I'_{[1]}$ in Figure 8. We get new intervals Z_{j+1} and J_{j+1} and the induction remains true by using the argument given for $j = 1$.

2) If $I_{[j+1]} \leq I'_{[j+1]}$, we couple according to the following procedure. The interval $I'_{[j+1]}$ is split into two intervals U and V such that $|V| = |I_{[j+1]}|$, which are coupled together.

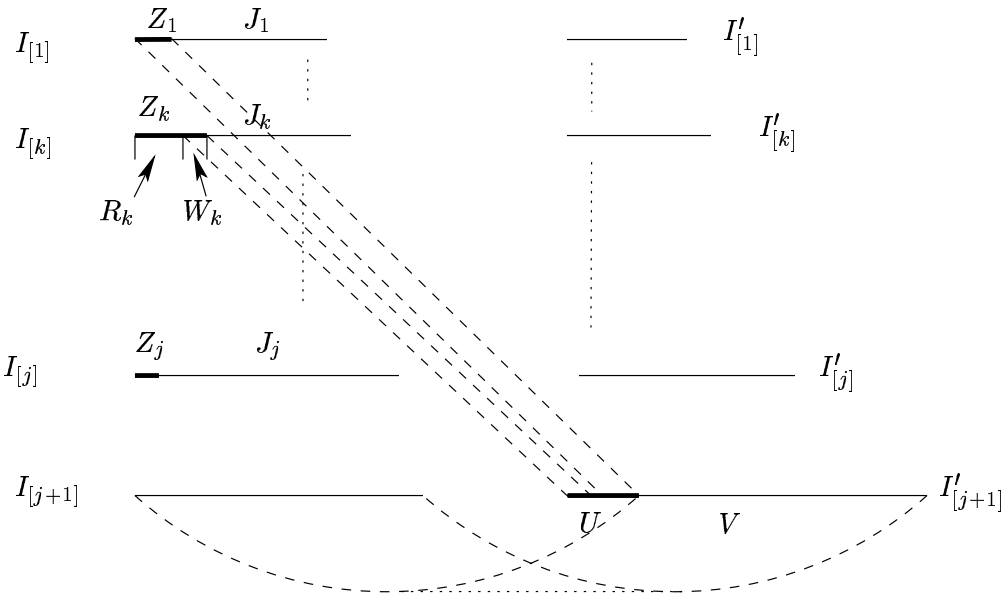

Figure 9: Coupling when $I_{[j+1]} \leq I'_{[j+1]}$.

Note that by the majorization property, $|U| = |I'_{[j+1]}| - |I_{[j+1]}| \leq |Z_1| + \cdots + |Z_j|$. Let $k := \min\{k : |Z_1| + \cdots + |Z_k| \geq |U|\}$. We split the interval Z_k into two intervals R_k, W_k such that $|W_k| = |U| - (|Z_1| + \cdots + |Z_{k-1}|)$. The coupling is illustrated in Figure 9.

- An error starting in V has the same probability to corrupt a frame than an error starting in $I_{[j+1]}$.

- An error starting in U has a smaller probability of corruption than an error starting in $Z_1 \cup \cdots Z_{k-1} \cup W_k$ because $|V| > |J_i|$ for all $i \leq k$.
- An error starting in $I'_{[1]} \cup \cdots \cup I'_{[j]}$ has a probability of corruption smaller or equal than an error starting in $J_1 \cup \cdots \cup J_j$ by the induction hypothesis.
- An error starting in $R_k \cup Z_{k+1} \cup \cdots \cup Z_j$ has a non-negative probability of corruption.

In total, $\mathbb{P}_{j+1} \geq \mathbb{P}'_{j+1}$.

Finally, the induction assumption is carried one more step by using the new splitting of $I_{[1]}, \cdots, I_{[j+1]}$ into

$$((J_1, \emptyset), \cdots (J_{k-1}, \emptyset), (J_k, R_k), (J_{k+1}, Z_{k+1}), \cdots, (J_j, Z_j), (I_{[j+1]} \cup Z_1 \cdots \cup Z_{k-1} \cup W_k, \emptyset)).$$

The proof is concluded by noticing that $\mathbb{P}_{one}(x) = \mathbb{P}_t + \mathbb{P}$(an error starts during a frame) and $\mathbb{P}_{one}(x') = \mathbb{P}'_t + \mathbb{P}$(an error starts during a frame) and using the result of the induction showing that $\mathbb{P}_t \geq \mathbb{P}'_t$ and the fact that the probability that an error starts during the transmission of a frame is equal in both cases. □

Theorem 3. *Under assumptions* $< A_1 >$ *and* $< A_2 >$ *and if* $\pi = 1$*, then for each FTU A, the optimal allocation* x_{one} *minimizing* $\mathbb{P}_{one}$ *groups together all replicas in the same round.*

Proof. Under $< A_2 >$ each burst can be considered independently because $\pi = 1$. Therefore, $\mathbb{P}_{one}$ is the sum of the loss probabilities for each burst. Let x be an arbitrary allocation. The restrictions over one round R of x and x_{one} are denoted $x|_R$ and $x_{one}|_R$ respectively. They obviously satisfy $I(x|_R) \prec I(x_{one}|_R)$. By periodicity $I(x) = (I(x|_R), I(x|_R), \cdots, I(x|_R))$ (repeated N times). This implies $I(x) \prec I(x_{one})$. Finally, applying Lemma 2 concludes the proof. □

This result calls for several comments.

- The assumption that a production cycle is a multiple of the size of a round is essential here, unlike in the proof of Lemma 1.
- If the assumption that $\pi = 1$ is removed then Lemma 2 is not true anymore. However, Theorem 3 may still be valid for some distributions of the burst size.

- The optimal allocation for $\mathbb{P}_{all}$ (that we call x_{all}) is Sturmian as proved in Theorem 2. As shown in [3], Sturmian sequences are minimal for the majorization ordering and therefore x_{all} has the worst possible performance according to $\mathbb{P}_{one}$. This shows that the two objectives (minimizing $\mathbb{P}_{one}$ and $\mathbb{P}_{all}$) are incompatible in a strong sense (the best allocation $\mathbb{P}_{one}$ is the worse for $\mathbb{P}_{all}$).

- Unlike for $\mathbb{P}_{all}$, the combined minimization of $\mathbb{P}_{one}$ for all FTUs is not a problem since the optimal solution groups all replicas of each FTU together.

5 Application to TTP/C

In this section we integrate the error handling mechanisms of TTP/C in the analysis developed so far and investigate in which extent the results remain valid.

5.1 TTP/C error handling mechanisms

The TTP/C protocol includes powerful but complex algorithms such as the clique avoidance and membership algorithms. In this paragraph, we give a simplified description of the functioning schemes of TTP/C version 1.0 that are related with transmission error handling and that might a priori interfere with our analysis. For instance, TTP/C defines the concept of "shadow" node. A shadow node replaces a defective node but does not possesses its own slot in the round. This redundancy scheme does not protect against transmission errors and we won't consider them in the rest of the paragraph.

1. Maximum Membership Failure Count (MMFC) check : if a node loses it membership in MMFC successive sending slots, then the controller terminates its operation by entering the "freeze state". It is an optional feature since MMFC can be set to zero which means no verification.

2. Lost of membership due to a incorrect transmission : if a frame is corrupted during its transmission the sender loses its membership and enters to the passive state. It waits in the passive state until it can re-acquire its slot. To re-acquire a slot the controller must have received the "minimum integration count" correct frames (should be set at least to a value of two), termed "the majority rule".

3. Re-integration of a node (transit from freeze state to passive state) : a "frozen" node must wait until the application sets the Controller On (CO) field to the

value "on". Then it must listen to a valid frame containing explicit C-state before entering the passive state. Then the node has to re-acquire its slot as described in point 2.

4. Clique avoidance algorithm : before starting to send a frame, a node must verify whether the number of frames that have been successfully sent in the last S slots (where S is the number of slots in the round so that it includes its own last transmission) is greater than the number of incorrect frames. In the latter case, the node enters the "freeze state" otherwise it transmits its frame and reset its counters. This rule will be termed the majority rule.

The rules 1,2 and 3 actually affect the value of $\mathbb{P}_{all}$ but not which allocation scheme is optimal. However, the majority rule of TTP/C (item 4 above) changes the combined minimization of $\mathbb{P}_{all}$ for all FTUs with respect to the general TDMA case. In fact, it makes it easier to reach optimal allocation for all FTUs together as shown in the next section.

As for the minimization of $\mathbb{P}_{one}$ no particular rule of TTP/C alters the results of Theorem 3.

5.2 Minimizing the loss probability

In the section, we assume $\pi = 1$: a bit covered by a perturbation will be corrupted with probability 1 and the whole frame is also corrupted. Under this assumption, it is possible to find an optimal allocation of all FTUs in the general case (without the condition on the cardinalities given in Section 3.2).

Here is the way to construction an optimal global allocation x_{all}. We construct two stacks S_1 and S_2 of slots. for each FTU i with C_i replicas, push $\lfloor C_i/2 \rfloor$ slots in the largest stack and $\lceil C_i/2 \rceil$ slots in the smallest stack.

The allocation x_{stack} is constructed by concatenating S_1 and S_2. The construction is illustrated by Figure 10.

Theorem 4. *Using TTP/C and under the foregoing assumptions, the allocation x_{stack} minimizes $\mathbb{P}_{all}$.*

Proof. the allocation x_{stack} has the following property: each FTU with more than two replicas has two replicas separated by at least $\lfloor S/2 \rfloor$ slots. Now, as soon as two replicas of the same message are allocated more that $\lfloor S/2 \rfloor$ slots apart, no perturbation can destroy both of them without freezing the system. This means that x_{stack} is optimal. □

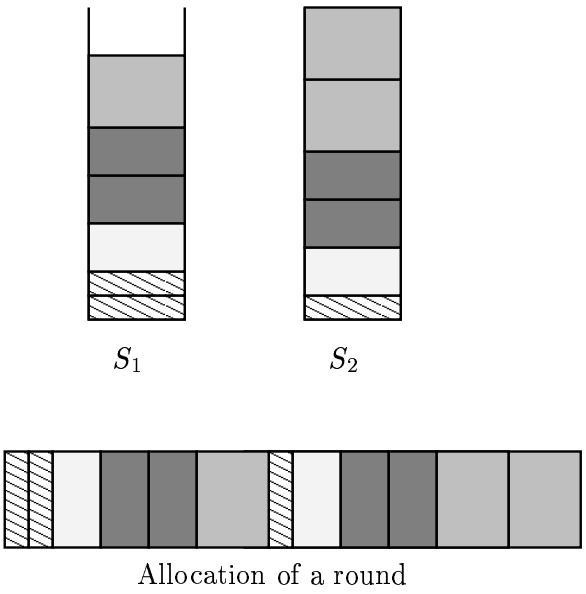

Figure 10: Construction of the optimal allocation x_{stack}.

Remark. With our error model, and because of the majority rule, it is useless the have more than two replicas per FTU if the objective is to minimize the corruption of all the replicas.

6 Concluding remarks

This study shows that for TDMA-based systems with bursty perturbations choosing the position of the replicas inside the round has a very important impact on the efficiency of the replication.

The first result of this study is to give an optimal way to spread the replicas in order to minimize the loss probability of all replicas. This result is valid when the cardinalities of the FTUs take up to two values or when the cardinalities are all powers of two. For the other cases, we provide a low-complexity heuristic which proves to be very efficient on the simulations that were performed.

In a second part, it has been proven that clustering together all replicas minimizes the probability to lose one or more replicas under a more general bursty perturbation model (the length of the bursts are not necessarily exponentially distributed). It has

been shown that the two objectives that were identified are incompatible in a strong sense.

In a future work, one may consider the case where a subset of FTUs requires the minimization of the loss probability while the rest of the FTUs need to minimize the probability that at least one replica is lost. This may be a situation arising on systems made of fail-silent and non fail-silent nodes. Another future work is to consider the use of Forward Error Correction techniques (such as Reed-Salomon codes) instead of replicas in order to make the system even more robust to transmission errors. Finally, we intend to study the robustness against transmission errors of an hybrid event-triggered/time-triggered network such as Flexray which is also considered for use in X-by-Wire automotive applications.

References

[1] E. Altman, B. Gaujal, and A. Hordijk. Admission control in stochastic event graphs. *IEEE Transaction on Automatic Control*, 45(5):854–868, 2000.

[2] E. Altman, B. Gaujal, and A. Hordijk. Multimodularity, convexity and optimization properties. *Mathematics of Operations Research*, 25(2):324–347, 2000.

[3] Eitan Altman, Bruno Gaujal, and Arie Hordijk. Regular ordering and applications in control policies. *Journal of Discrete Event Dynamic Systems*, 12(2):187–210, 2002.

[4] J. Barrenscheen and G. Otte. Analysis of the physical CAN bus layer. In 4^{th} *international CAN Conference, ICC'97*, pages 06.02–06.08, Octobre 1997.

[5] E. Dilger, T. Führer, B. Müller, and S. Poledna. The x-by-wire concept: Time-triggered information exchange and fail silence support by new system services. Technical Report 7/1998, Technische Universität Wien, Institut für Technische Informatik, 1998. also available as SAE Technical Paper 98055.

[6] B. Gaujal and N. Navet. Traffic shaping in real-time distributed systems: a low-complexity approach. *Computer Communications*, 22(17):1562–1573, 1999.

[7] G. Grünsteidl, H. Kantz, and H. Kopetz. Communication reliability in distributed real-time systems. In *10th Workshop on Distributed Computer Control Systems*, 1991.

[8] B. Hajek. Extremal splittings of point processes. *Mathematics of Operation Research*, 10(4):543–556, 1985.

[9] International Standard Organization ISO. *Road Vehicles - Low Speed serial data communication - Part 2: Low Speed Controller Area Network*. ISO, 1994. ISO 11519-2.

[10] H. Kopetz. *Real-Time Sytems : Design Principles for Distributed Embedded Applications*. Kluwer Academic Publishers, Boston, 1997.

[11] H. Kopetz, G. Bauer, and S. Poledna. Tolerating arbitrary node failures in the time-triggered architecture. In *SAE 2001 World Congress, March 2001, Detroit, MI, USA*, Mar. 2001.

[12] H. Kopetz, R. Nossal, R. Hexel, A. Krüger, D. Millinger, R. Pallierer, C. Temple, and M. Krug. Mode handling in the time-triggered architecture. *Control Engineering Practice*, 6(1998):61–66, Mar. 1998.

[13] A. W. Marshall and I. Olkin. *Inequalities: Theory of Majorization and its Applications*, volume 143 of *Mathematics in Science and Engineering*. Academic Press, 1979.

[14] N. Navet, Y.-Q. Song, and F. Simonot. Worst-case deadline failure probability in real-time applications distributed over CAN (Controller Area Network). *Journal of Systems Architecture*, 46(7):607–618, 2000.

[15] I.E. Noble. EMC and the automotive industry. *Electronics & Communication Engineering Journal*, pages 263–271, Octobre 1992.

[16] OSEK Group. OSEK/VDX fault-tolerant communication, July 2001. Version 1.0, available at http://www.osek-vdx.org/.

[17] H. Pfeifer. Formal verification of the ttp group membership algorithm. In *FORTE/PSTV 2000*, 2000.

[18] Stefan Poledna, Peter Barrett, Alan Burns, and Andy Wellings. Replica determinism and flexible scheduling in hard real-time dependable systems. *IEEE Transactions on Computers*, 49(2):100–111, Feb. 2000.

[19] C. Temple. Avoiding the babbling-idiot failure in a time-triggered communication system. In *International Symposium on Fault-Tolerant Computing (FTCS)*, pages 218–227, 1998.

[20] TTTech Computertechnik GmbH. *Specification of the TTP/C Protocol - version 1.0*, July 2002.

[21] E. Zanoni and P. Pavan. Improving the reliability and safety of automotive electronics. *IEEE Micro*, 13(1):30–48, 1993.

Unit´e de recherche INRIA Lorraine, Technopˆole de Nancy-Brabois, Campus scientifique,
615 rue du Jardin Botanique, BP 101, 54600 VILLERS LÈS NANCY
Unit´e de recherche INRIA Rennes, Irisa, Campus universitaire de Beaulieu, 35042 RENNES Cedex
Unit´e de recherche INRIA Rhˆone-Alpes, 655, avenue de l'Europe, 38330 MONTBONNOT ST MARTIN
Unit´e de recherche INRIA Rocquencourt, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex
Unit´e de recherche INRIA Sophia-Antipolis, 2004 route des Lucioles, BP 93, 06902 SOPHIA-ANTIPOLIS Cedex

Éditeur
INRIA, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex (France)
http://www.inria.fr
ISSN 0249-6399